# I3RC Monte Carlo Model

*Michael J. Garay and Roger Davies*
*University of Arizona*
*Institute of Atmospheric Physics*
*Tucson, AZ*

## Introduction

The University of Arizona's Monte Carlo model used in the intercomparison of 3D radiative transfer codes (I3RC) is a basic, forward model which involves no photon weighting, simplifying assumptions or photon history truncation. In the model each photon is tracked from its beginning at the top of the region of interest until it is either absorbed or it exits the region and is counted. The primary purpose of this model is to provide continuity between the finite cloud results of *Davies* [1976] and *Davies* [1978] and the results of more recent models.

## Model Description

Our present Monte Carlo code is, for the most part, an updated FORTRAN 90 version of the original FORTRAN 77 code used by *Davies* [1976] and *Davies* [1978] for modeling radiative transfer through isolated three-dimensional "cubic" clouds. We tested the performance and accuracy of the FORTRAN 90 version by direct comparison with the results of *Davies* [1976] for the same isolated cloud cases. Additionally, because computer time was expensive when the original code was developed, the FORTRAN 77 code was optimized to run as quickly and efficiently as possible. Subtle coding "tricks," such as minimizing the use of the *IF-THEN-ELSE* construct through the use of *GOTO*s and obtaining the sine of an angle using the formula $\sin\theta = \sqrt{1-\cos^2\theta}$, were retained in the present version. Thus, while the structure of our FORTRAN 90 code may not appeal to computer programmers, our code is likely to be among the most highly optimized for performance in the intercomparison.

A second subtlety which deserves some mention is the choice of the random number generator. For a large number of trials such as occur in a Monte Carlo run, the choice of a good (pseudo)random number generator is important in order to eliminate biases in the results. In our case, the random number generator *RAN2* from *Numerical Recipes in Fortran 77* [1992] was chosen for its simplicity and ease of coding. This random number generator was tested in the Monte Carlo routine by sending photons into the top of a cubic cloud whose scattering was specified to be isotropic. Any biases coming from the choice of random numbers would, of course, appear in the resultant statistics for the number of photons coming out each of the four cloud sides. The expected error in the photon flux, $F$, is given theoretically as $\sigma = \sqrt{F(1-F)/N}$, where $N$ is the number of Monte Carlo trials. In this test, *RAN2* yielded statistical errors far below this theoretical threshold and, given the simplicity of *RAN2*, it is likely that nearly all commonly available random number generators perform well enough for use in Monte Carlo codes. Therefore the choice of the random number generator becomes one of personal preference and algorithm speed.

Another subtlety, which plays an important role in the Monte Carlo method, is the treatment of the scattering phase function. A computationally efficient way of dealing with the phase function is to generate a "look-up" table of the integrated and normalized phase function values using code separate from that used for the Monte Carlo routine itself. This table will contain values from zero to one and the associated scattering angle, whose occurrence is weighted by the probability of a photon being scattering in that particular direction. Thus, when a random number between zero and one is generated by the Monte Carlo routine, the scattering angle can be found by locating the appropriate number in the look-up table. However, in dealing with sharply peaked phase functions, such as the Henyey-Greenstein used in the intercomparison, it is important to adequately capture the details of the scattering peak. Davies was able to do this for the double Henyey-Greenstein phase functions he used with a mere 200 tabulated values, or "bins," spaced equally in probability space. A less elegant method was used to generate the look-up tables for the present code. Instead of spacing the bins equally in probability, a more direct approach, which spaced the bins equally in the cosine of the scattering angle, was used. This parameterization was found to be superior to spacing the bins equally in scattering angle itself, however. Even so, to match the results given in *Davies* [1976] for the double Henyey-Greenstein within the expected error of the Monte Carlo trials 10,000 bins equal in the cosine of the scattering angle were necessary. However, because the Monte Carlo code "jumps" directly to the bin in the scattering lookup table instead of looping through the values in the table to make a match, the issue becomes merely one of storage space, rather than speed. Still, the comparison with Davies' results suggests that care must be taken in the way the phase function is parameterized.

## Application to I3RC

Since the original code was designed to be run on isolated, "cubic" clouds of various optical depths, it was quite easy to adapt the code to run on the three test cases provided. Each cloud field in the intercomparison was initialized as an array of "cubes," each with its own extinction parameter. However, since we had been working with analytic phase functions, the C1 phase function, as provided for the intercomparison, was not used due to insufficient time to update the look-up tables. In spite of this, results were submitted for all parts of Case 1, the two-street thick-thin step-function cloud, and those parts of Case 2, the ARM Radar reconstructed cloud, not involving the C1 phase function.

## Results

The results of Phase 1 of the intercomparison have been submitted for all cases for which results were obtained. These include results for the two-street thick-thin step-function cloud for all sun angles and single scatter albedos. Our results appear to agree well with those obtained by other groups in the intercomparison. However, discrepancies between our results and those of other groups exist, even for this simple case, which lie outside the range of the theoretically expected errors in the Monte Carlo simulation. For the ARM Radar reconstructed cloud, an error in correctly counting the horizontal transport of photons scattered from the underlying surface, in the case where the surface was present, gave a nonsensical result, but *only* for the horizontal flux and *only* in this case. Finally, constraints on available computer time prevented submission of results for the marine boundary layer cloud from Landsat case. Due to the extent of the cloud field, the number of photons trials generally being made by the Monte Carlo ($10^8$) was inadequate to capture the details of the radiation.

## Discussion

A glance through the resultant mean values of the two-street thick-thin step-function cloud reveals a consistent trend between our results and the results of other groups. This same trend is much more pronounced for the ARM Radar reconstructed cloud. Our values for mean albedo tend to be low relative to many of the other Monte Carlo models, while our values for mean transmittance tend to be high. Our absorptance and reflectivity values, for the most part show good agreement, however.

Although conjectural, it is our belief that these differences are due to the phase function parameterization within the Monte Carlo model, rather than any errors present in the code itself.

First of all, our Monte Carlo code was tested extensively against the results of *Davies* [1976] for all manner of optical depths and directional illumination. While the kernel of the Monte Carlo code is identical to that used by Davies, this comparison still allowed us to identify any major computational errors present before participating in the intercomparison. Because of these tests, we are fairly confident that there are no significant errors still present in the code. It is likewise puzzling that we do well relative to the other models, for the most part, in calculating reflectivities, but fail to obtain consensus values for the fluxes. This is particularly interesting since the code was exhaustively run against *Davies* [1976] results for *fluxes*, specifically, and for cases with much larger total cloud optical depths (and therefore more scattering events) with a more complicated phase function, which included a backward peak.

After the results from Phase 1 became available a test was run using our Monte Carlo code for simple cubic clouds which involved changing the parameterization of the single Henyey-Greenstein phase function. The results are summarized below:

| **Tau = 10** | | | **Tau = 20** | | | **Tau = 50** | | |
|---|---|---|---|---|---|---|---|---|
| Bins | Albedo | Transmit | Bins | Albedo | Transmit | Bins | Albedo | Transmit |
| 100 | 0.1196 | 0.1473 | 100 | 0.2236 | 0.0576 | 100 | 0.4247 | 0.0170 |
| 1000 | 0.1044 | 0.1751 | 1000 | 0.1999 | 0.0706 | 1000 | 0.3928 | 0.0203 |

This table shows that for bins spaced equally in the cosine of the phase function, more bins decreases the albedo and increases the transmittance within our Monte Carlo code. This corresponds exactly to the differences between our results (which use 10,000 bins, equal in cosine of the scattering angle) and those of the other Monte Carlo groups. However, not knowing the details of the parameterizations used by the other groups, we can only say this is a likely explanation of the differences. For reflectivities, there is less dependence on the details of the phase function, so this further explains why our results compare favorably with the other groups for some of the reflectivities. However, in the case of strong forward scattering, we see that our results once again differ in the reflectivities from those of the other groups and this, too, can be explained by our more accurate treatment of the forward peak of the scattering phase function.

The results of Phase 1, at least from the perspective of our Monte Carlo, model suggest two important issues which should be dealt with in Phase 2. The first of these is the way the phase function is parameterized in the Monte Carlo models. Our results show that it is difficult to made head-to-head comparisons between models using different parameterizations of the scattering phase function because this parameterization has a large effect on the outcome of the Monte Carlo trials. Secondly, in order for the Monte Carlo models to serve as any sort of "benchmark," they should be tested against a case for which the analytical solution is known. During the workshop, most groups seemed confident in their ability to handle the "pedagogical" case of the two-street thick-thin step-function cloud, but the intercomparison results show many differences, even among the Monte Carlo models, which cannot be explained by the expected errors in the methods. While in some sense it might be "boring" to run the models on a exceedingly simple case, the results of such a test might very well be enlightening as to the fundamental difficulties which must be addressed before the Monte Carlo models can be used for comparison purposes to other 3D radiative transfer methods.

## References

Davies, R., 1976: *The three-dimensional transfer of solar radiation in clouds.* Ph.D. thesis, University of Wisconsin, 220 pp.

Davies, R., 1978: The effect of finite geometry on the three-dimensional transfer of solar irradiance in clouds. *J. Atmos. Sci.*,**35**, 1712-1724.

Press, W. H., S. A. Teukolsky, W. T. Vetterling, William T. and B. P. Flannery, 1992: *Numerical Recipes in FORTRAN 77 Second Ed.*, Cambridge University Press, 992 pp.